What Is Claimed Is:

1. An apparatus that performs modular division, comprising:

a register *A* that is initialized with a value `X`;

a register *U* that is initialized with a value `Y`;

a register *B* that is initialized with a value `M`;

a register *V* that is initialized with a value `0`;

a counter *CA* that indicates an upper bound for the most-significant non-zero bit of register *A*;

a counter *CB* that indicates an upper bound for the most-significant non-zero bit of register *B*; and

an updating mechanism that is configured to iteratively reduce the contents of registers *A* and *B* to a value of one by applying a plurality of invariant operations to registers *A*, *B*, *U* and *V*;

wherein updating mechanism is configured to use the counters *CA* and *CB* to estimate the relative magnitudes of the values stored in registers *A* and *B* instead of performing an expensive comparison operation between register *A* and register *B*.

2. The apparatus of claim 1, further comprising:

a temporary register *H*; and

a temporary register *L*;

wherein updating mechanism is configured to temporarily store $A + B$ in the temporary register *H*; and

wherein updating mechanism is configured to temporarily store $U + V$ in the temporary register *L*.

3. The apparatus of claim 1, wherein the initial values in the registers A, B, U and V satisfy invariant relationships.

4. The apparatus of claim 3, wherein the invariant relationships include:

$A*\mathtt{Y} = U*\mathtt{X} \bmod \mathtt{M}$; and

$B*\mathtt{Y} = V*\mathtt{X} \bmod \mathtt{M}$.

5. The apparatus of claim 4, wherein the updating mechanism is configured to maintain the invariant relationships between the registers A, B, U and V after application of the plurality of invariant operations.

6. The apparatus of claim 5, wherein the plurality of invariant operations comprise:

if A is even and U is even, then $A{:=}SHIFT(A)$, $U{:=}SHIFT(U)$, $CA{:=}CA\text{-}1$;

if A is even and U is odd, then $A{:=}SHIFT(A)$, $U{:=}SHIFT(U+M)$, $CA{:=}CA\text{-}1$;

if B is even and V is even, then $B{:=}SHIFT(B)$, $V{:=}SHIFT(V)$, $CB{:=}CB\text{-}1$;

if B is even and V is odd, then $B{:=}SHIFT(B)$, $V{:=}SHIFT(V+M)$, $CB{:=}CB\text{-}1$;

if $CA>CB$, then $A{:=}A+B$ and $U{:=}U+V$; and

if $CA \leq CB$, then $B{:=}A+B$ and $V{:=}U+V$;

wherein the *SHIFT* operation denotes a right shift by one bit of the register contents.

7. The apparatus of claim 6,

wherein setting $A = A + B$ and $U = U + V$ involves first setting $H = A + B$ and $L = U + V$, and later setting $A = H$ and $U = L$ if $CA \geq CB$; and

wherein setting $B = A + B$ and $V = U + V$ involves first setting $H = A + B$ and $L = U + V$, and later setting $B = H$ and $V = L$ if $CA < CB$.

8. The apparatus of claim 7, wherein the operations of setting $H = A + B$, setting $L = U + V$, and determining if $CA \geq CB$ or if $CA < CB$ take place concurrently.

9. The apparatus of claim 1, wherein components of the updating mechanism operate asynchronously, without use of a centralized clock signal.

10. An apparatus that performs modular division, comprising:

a register A that is initialized with a value X;

a register U that is initialized with a value Y;

a register B that is initialized with a value M;

a register V that is initialized with a value 0;

wherein the initial values in the registers A, B, U and V satisfy invariant relationships, including, A*Y = U*X mod M, and B*Y = V*X mod M;

a temporary register H;

a temporary register L;

a counter CA that indicates an upper bound for the most-significant non-zero bit of register A;

a counter CB that indicates an upper bound for the most-significant non-zero bit of register B; and

an updating mechanism that is configured to iteratively reduce the contents of one of the counters CA and CB to a value less than zero by applying a plurality of invariant operations to registers A, B, U and V;

wherein the updating mechanism is configured to maintain the invariant relationships between the registers A, B, U and V after application of the plurality of invariant operations;

wherein updating mechanism is configured to temporarily store $A + B$ in the temporary register H;

wherein updating mechanism is configured to temporarily store $U + V$ in the temporary register L;

wherein the updating mechanism is configured to use the counters CA and CB to estimate the relative magnitudes of the values stored in registers A and B instead of performing an expensive comparison operation between register A and register B.

11. The apparatus of claim 10, wherein the plurality of invariant operations comprise:

if A is even and U is even, then $A:=SHIFT(A)$, $U:=SHIFT(U)$, $CA:=CA-1$;

if A is even and U is odd, then $A:=SHIFT(A)$, $U:=SHIFT(U+M)$, $CA:=CA-1$;

if B is even and V is even, then $B:=SHIFT(B)$, $V:=SHIFT(V)$, $CB:=CB-1$;

if B is even and V is odd, then $B:=SHIFT(B)$, $V:=SHIFT(V+M)$, $CB:=CB-1$;

if $CA>CB$, then $A:=A+B$ and $U:=U+V$; and

if $CA\leq CB$, then $B:=A+B$ and $V:=U+V$;

wherein the $SHIFT$ operation denotes a right shift by one bit of the register contents.

12. The apparatus of claim 11,

wherein setting $A = A + B$ and $U = U + V$ involves first setting $H = A + B$ and $L = U + V$, and later setting $A = H$ and $U = L$ if $CA \geq CB$; and

wherein setting $B = A + B$ and $V = U + V$ involves first setting $H = A + B$ and $L = U + V$, and later setting $B = H$ and $V = L$ if $CA < CB$.

13. The apparatus of claim 12, wherein the operations of setting $H = A + B$, setting $L = U + V$, and determining if $CA \geq CB$ or if $CA < CB$ take place concurrently.

14. The apparatus of claim 10, wherein components of the updating mechanism operate asynchronously, without use of a centralized clock signal.

15. A method for performing modular division, comprising:

initializing a register A with a value X;

initializing a register U with a value Y;

initializing a register B with a value M;

initializing a register V with a value 0;

maintaining a counter CA that indicates an upper bound for the most-significant non-zero bit of register A;

maintaining a counter CB that indicates an upper bound for the most-significant non-zero bit of register B; and

iteratively reducing the contents of registers A and B to a value of one by applying a plurality of invariant operations to registers A, B, U and V;

wherein applying the plurality of invariant operations involves using the counters CA and CB to estimate the relative magnitudes of the values stored in registers A and B instead of performing an expensive comparison operation between register A and register B.

16. The method of claim 15, wherein iteratively reducing the contents of registers A and B involves:

temporarily storing $A + B$ in a temporary register H; and

temporarily storing $U + V$ in a temporary register L.

17. The method of claim 16, wherein the initial values in the registers A, B, U and V satisfy invariant relationships.

18. The method of claim 17, wherein the invariant relationships include:

$A*\mathtt{Y} = U*\mathtt{X} \bmod \mathtt{M}$; and

$B*\mathtt{Y} = V*\mathtt{X} \bmod \mathtt{M}$.

19. The method of claim 18, wherein applying the plurality of invariant operations involves maintaining the invariant relationships between the registers A, B, U and V.

20. The method of claim 19, wherein the plurality of invariant operations comprise:

if A is even and U is even, then $A{:=}SHIFT(A)$, $U{:=}SHIFT(U)$, $CA{:=}CA\text{-}1$;

if A is even and U is odd, then $A{:=}SHIFT(A)$, $U{:=}SHIFT(U+M)$, $CA{:=}CA\text{-}1$;

if B is even and V is even, then $B{:=}SHIFT(B)$, $V{:=}SHIFT(V)$, $CB{:=}CB\text{-}1$;

if B is even and V is odd, then $B{:=}SHIFT(B)$, $V{:=}SHIFT(V+M)$, $CB{:=}CB\text{-}1$;

if $CA>CB$, then $A{:=}A+B$ and $U{:=}U+V$; and

if $CA\leq CB$, then $B{:=}A+B$ and $V{:=}U+V$;

wherein the *SHIFT* operation denotes a right shift by one bit of the register contents.

21. The method of claim 20,

wherein setting $A = A + B$ and $U = U + V$ involves first setting $H = A + B$ and $L = U + V$, and later setting $A = H$ and $U = L$ if $CA \geq CB$; and

wherein setting $B = A + B$ and $V = U + V$ involves first setting $H = A + B$ and $L = U + V$, and later setting $B = H$ and $V = L$ if $CA < CB$.

22. The method of claim 21, wherein the operations of setting $H = A + B$, setting $L = U + V$, and determining if $CA \geq CB$ or if $CA < CB$ take place concurrently.

23. The method of claim 15, wherein operations involved in performing the method take place asynchronously, without use of a centralized clock signal.